

4 Telecom Interoperability Issues You Can Easily Solve with a Custom SBC

Interoperability failures are the silent profit killers of modern voice networks. They show up as dropped calls, one-way audio, and failed transfers. Engineering teams spend weeks chasing 488 responses and codec mismatches. Off-the-shelf Session Border Controllers solve common cases. But multi-vendor, multi-carrier, and Voice AI deployments quickly expose their limits.

The global SBC market reached USD 755 million in 2025 and is projected to hit USD 1.24 billion by 2034. That growth is driven by 5G rollout, BYOC adoption, and Voice AI integrations. Each layer adds new SIP dialects and codec demands. A custom SBC, built on Kamailio, FreeSWITCH, or OpenSIPS, gives your team full control. Header rewriting, transcoding logic, and routing decisions become engineering choices.

Here are the four interoperability issues a custom SBC actually solves.

Why Telecom Interoperability Still Breaks in 2026

SIP is a standard. SIP implementations are not. Every PBX vendor, every carrier, and every UCaaS platform speaks its own dialect. Headers differ. DTMF methods differ. Codec offers differ. Session timer behavior differs.

The interop surface has expanded sharply over the last three years. BYOC for Microsoft Teams and Zoom Phone added new certification paths. STIR/SHAKEN enforcement reshaped North American carrier traffic. Voice AI agents introduced Opus 16 kHz media flows alongside legacy G.711.

Stock SBC vendor profiles handle the common combinations. They struggle when a new carrier ships a non-standard P-Asserted-Identity format. They charge per channel for transcoding licenses. They wait quarters to certify your next platform. A custom SBC moves these decisions inside your engineering team.

This is exactly where our [custom session border controller engineering work](#) sits. The four issues below are the ones ~~we see most often in production deployments.~~

Issue 1: SIP Dialect Mismatch Between PBX and Carrier

What it looks like in production

Calls fail at INVITE. Carriers return 488 Not Acceptable. From headers carry the wrong domain. Diversion headers are missing on forwarded calls. P-Asserted-Identity arrives with format the upstream rejects.

These are the classic interop symptoms. Different PBX vendors implement SIP headers, DTMF, and codec negotiation differently. The result is calls that pass a lab test and fail in production.

Why off-the-shelf SBCs hit a wall

Vendor SBCs ship with per-trunk-group normalization profiles. These cover Twilio, Bandwidth, and a list of certified PBX systems. Add a new regional carrier, and you wait for a vendor update. Some platforms expose a scripting layer. Most charge extra for it.

How a custom SBC fixes it

Kamailio and OpenSIPS expose full SIP header manipulation as native scripting. You can rewrite From, To, Contact, Diversion, History-Info, and P-Asserted-Identity on any leg. Rules apply per trunk group, per route, or per call. Regex matching gives surgical control over edge cases.

A custom SBC also lets you handle DTMF method translation. Some carriers send DTMF as RFC 2833. Some send it as SIP INFO. Older endpoints send it inband. The SBC converts between methods so IVRs and contact centers work consistently.

Real example

We worked on a deployment where a legacy Avaya PBX needed to talk to a STIR/SHAKEN enforced US carrier. The carrier rejected calls without proper attestation headers. A custom Kamailio routing block added the headers, signed where required, and passed verification status downstream. Lab to production cutover took two weeks.

Issue 2: Codec Mismatch and Transcoding Failures

The symptom

One-way audio. Calls that drop at the 30 second mark. 488 responses citing no common codec. Polycom bridges offering 21 codecs that crash the upstream SIP stack.

These failures trace back to codec negotiation. A mobile carrier offering AMR-WB and an enterprise PBX supporting only G.711 will fail to negotiate unless the SBC transcodes between them.

Common codec combat zones

- G.711 versus G.729 Annex B (DTX behavior breaks calls)
- AMR-WB on mobile carriers versus narrowband G.711 on enterprise PBX
- Opus on WebRTC versus G.711 on legacy SIP trunks
- G.722 wideband on Polycom versus G.711 on contact center platforms

What custom SBC transcoding adds

FreeSWITCH and its mod_sofia module handle real-time transcoding at scale. Hardware acceleration through Sangoma or Intel QuickAssist cards reduces CPU load. Per-route codec policy means you choose what gets transcoded and what passes through. No blanket per-channel licensing.

A custom SBC also restricts codec offers. You can trim a 21 codec SDP down to four supported formats. You can force G.711 on routes that hit fax endpoints. You can prefer Opus on WebRTC and downgrade on the carrier leg.

Voice AI angle

Voice AI workloads need wideband audio. Speech-to-text models perform best with 16 kHz Opus or PCM. Telephony carriers ship 8 kHz G.711. A custom SBC bridges the two. It transcodes the carrier leg to Opus 16 kHz for the AI agent, then back to G.711 for the call recording archive. Without that bridge, sub-600ms voice AI latency is unreachable on Twilio SIP.

Issue 3: NAT Traversal and Topology Exposure

The symptom

SIP registration succeeds. Audio fails. Remote softphones behind carrier grade NAT cannot send media. WebRTC endpoints work in the office and break on cellular. Internal IP addresses leak in SIP and SDP, exposing your network topology to upstream carriers.

Where canned SBCs underperform

Most stock SBCs do basic STUN and TURN. Few let you tune media anchoring policy per route. Topology hiding is often a binary toggle, not a configurable rewrite. ICE integration with WebRTC clients varies by vendor.

Custom SBC controls

- Selective media anchoring (anchor only when NAT detected, pass through otherwise)
- Native ICE and TURN integration with WebRTC gateways
- RTP latching for endpoints behind symmetric NAT
- Full topology hiding through B2BUA re-origination
- Header scrubbing for Via, Record-Route, and internal SDP candidates

Bonus, compliance

Topology hiding is also a compliance control. Financial services and healthcare clients require internal network maps stay private. A custom SBC strips internal IPs from every outbound message. Upstream carriers see only the SBC public interface.

Issue 4: Multi-Carrier Routing, Failover, and Least-Cost Decisions

The symptom

The primary carrier degrades. Calls fail before manual switching kicks in. There is no real-time quality-based routing. Least-cost routing depends on a static dial plan that nobody has updated in eight months.

Why vendor SBC routing engines fall short

Stock SBC routing tables are static. Some platforms support external lookups but cap query rate. Real-time MOS-aware routing requires features that sit behind expensive licensing tiers.

Custom SBC routing capabilities

- HTTP and REST lookups to external rating engines on each call
- MOS-aware dynamic rerouting based on live call quality metrics
- CDR-driven least-cost routing with per-destination thresholds
- Carrier failover on 503, timeout, or 5xx response patterns
- STIR/SHAKEN attestation-level routing for compliance-sensitive traffic

Use case

An MSP routes US termination across three SIP trunks. Each call queries an external rating engine. The SBC weighs cost, ASR over the last 15 minutes, and STIR/SHAKEN attestation level. Calls to high-risk destinations route only through carriers offering A-level attestation. Cost savings averaged 18 percent in the first quarter.

Off-the-Shelf vs Custom SBC: Where Each Fits

Both deployment paths have a place. The decision comes down to how much interoperability work your team faces and how often.

Factor	Off-the-Shelf SBC	Custom SBC
Time to integrate a new carrier	Wait for vendor profile update	Days, controlled by your team
Transcoding licensing	Per channel, recurring	One-time build, hardware scale
Header manipulation	Fixed profiles, limited regex	Full scripting, per-call logic
Voice AI integration	Limited, codec gaps	Native Opus 16 kHz bridging

Scaling cost per channel	Steep at high volumes	Flatter, infrastructure driven
Best fit	One or two carriers, standard PBX	CPaaS, ITSP, MSP, Voice AI vendors

Off-the-shelf wins on speed for simple deployments. Custom wins on flexibility and unit economics when call volume grows.

What a Custom SBC Actually Looks Like Under the Hood

A custom SBC is rarely written from scratch. It is built on proven open-source foundations and extended with your business logic.

Core open-source components

- Kamailio for high-throughput SIP signaling and routing logic
- FreeSWITCH for media handling, transcoding, and call control
- OpenSIPS for carrier-grade SIP load balancing and load distribution
- RTPengine for media proxy and topology hiding

Custom layers we typically add

- REST APIs for provisioning and tenant management
- Real-time analytics and CDR pipelines
- Fraud detection hooks and STIR/SHAKEN signing or verification
- Multi-tenant isolation for CPaaS and white-label use cases
- Voice AI gateway modules for STT, TTS, and NLU integration

Deployment models

Bare metal, virtualized, containerized on Kubernetes, or cloud-native on AWS and Azure. The right model depends on call volume, latency requirements, and your DevOps maturity.

Do You Actually Need a Custom SBC? A 5-Point Check

Custom is not for every deployment. Use this check before committing.

- You connect to five or more upstream carriers or PBX vendors
- Your team opens more than two interop tickets per month
- Transcoding licenses are a meaningful line item in your bill
- You need Teams Direct Routing, Zoom BYOC, and STIR/SHAKEN under one roof
- Voice AI is on your 12-month roadmap, with sub-600ms latency targets

If three or more apply, a custom SBC build pays back inside 12 months. If one or two apply, extending an open-source SBC is usually enough.

Closing Thoughts

Interoperability is where voice deployments live or die. The teams that win invest in tools they control. At Ecosmob, we build custom SBC stacks for telecom operators, CPaaS providers, MSPs, and Voice AI platforms. We work across Kamailio, FreeSWITCH, OpenSIPS, and RTPengine, and we ship to production. Explore our [custom VoIP development services](#) or learn more about our session border controller solutions. If you are evaluating a build, [contact us](#) for a scoping discussion.

Frequently Asked Questions

What is the difference between a SIP proxy and a session border controller?

A SIP proxy forwards requests without terminating the call. A session border controller uses a B2BUA architecture and fully terminates and re-originates both signaling and media legs. That gives the SBC full control over headers, media, and security.

Can a custom SBC handle Teams Direct Routing and Zoom BYOC together?

Yes. The SBC presents two trunk groups, one toward Teams and one toward Zoom. Each gets its own normalization profile. The same SBC can also expose SIP trunks toward your enterprise PBX systems on the south side.

How does a custom SBC solve one-way audio across NAT?

Through media anchoring, RTP latching, and ICE integration. The SBC re-originates the media leg from its own public interface. Endpoints behind NAT exchange RTP with the SBC instead of each other. RTPengine is the typical media engine for this.

Is Kamailio or FreeSWITCH the better base for a custom SBC?

Kamailio handles signaling at high throughput, often 10,000 calls per second on commodity hardware. FreeSWITCH handles media, transcoding, and call control. Most production custom SBCs run both, with Kamailio at the edge and FreeSWITCH for media processing.

How long does a custom SBC build typically take?

A focused build with two or three trunk integrations, basic routing, and transcoding takes 8 to 12 weeks. Adding multi-tenant provisioning, fraud detection, and Voice AI bridging pushes timelines to 16 to 24 weeks. Hardware sizing and load testing add another two to four weeks.

Does a custom SBC support STIR/SHAKEN attestation?

Yes. Kamailio and OpenSIPS both support STIR/SHAKEN signing and verification through native modules. Custom logic can route based on attestation level, sign outbound calls with your certificate, and reject inbound calls below your minimum threshold.